



Creando un pool de conexiones

Transcripción

[00:00] Hola, ya revisamos el proyecto y planteamos los próximos pasos para dejar nuestra aplicación más completa y siguiendo las prácticas del mercado. Ahora vamos a implementar el pool de conexiones para que nuestra aplicación tenga un mejor control de las mismas, sin comprometer tanto la experiencia de los usuarios como la base de datos.

[00:19] Lo primero que vamos a hacer es agregar el driver de C3P0 a nuestra aplicación. Y acá desde que empezamos con el proyecto estamos tratando la importación de librerías como dependencias del Maven, y con esta no va a ser distinto.

[00:33] Acá en Eclipse nosotros vamos a abrir el archivo pom.xml para agregar esta dependencia de pool llamado C3P0, y junto con ella vamos a agregar una dependencia más también que se llama mchange como Java. Ya les muestro cómo va a quedar el Código.

[00:49] Acá en la tag de dependencies, en donde agregamos todas las dependencias, nosotros vamos a hacer la dependency, agregamos una acá y la primera va a ser la de C3P0. Entonces su groupId va a ser com.mchange, ese es el groupId, y el artifactId va a ser c3p0, así. Y la versión que vamos a estar utilizando en este curso es la 0.9.5.5. Listo, esta es la dependencia de C3P0.

[01:30] La otra dependencia que vamos a agregar es esta de acá, dependency y ahí ponemos el groupId, que va a ser también com.mchange. El artifactId va a

ser mchange-commons-java y la versión va a ser la 0.2.20. Okay, ¿pero qué son estas dependencias?

[02:04] Bueno, la primera acá, la de C3P0, nos va a dar la posibilidad de configurar un pool de conexiones y esa segunda del mchange-commons-java nos va a dar la posibilidad de agregar más detalles del datasource vía el log de la aplicación en la consola.

[02:22] Bueno, cuando con las dependencias agregadas ahora, nosotros vamos a la clase connectionFactory para realizar los cambios necesarios para la configuración del pool de conexiones. La idea acá es que la connectionFactory utilice las conexiones desde el pool de conexiones, en lugar de salir creando una conexión de forma pura, como estamos haciendo ahora.

[02:42] Bueno, aquí en la connectionFactory entonces vamos a crear un constructor connectionFactory acá, pongo el nombre de la clase y ya me aparece la ayuda para el constructor. Y acá en el constructor nosotros vamos a crear una variable que va a hacer el pooledDataSource. ¿Y qué va a recibir esta variable? Va a ser un new de ComboPooledDataSource();

[03:11] A ver si ya se hizo el auto complete y ahí está. Esta clase es la de C3P0. En esta instancia del pool nosotros vamos a estar configurando la URL de la base de datos, el usuario y la contraseña, tal como estamos haciendo con la conexión. Entonces yo voy a estar copiando acá esta variable para que sea más rápido y con el pooledDataSource yo voy a hacer un setJdbcUrl. ¿En dónde?

[03:37] Acá voy a copiar esta URL que ya tenemos y la pegamos acá. La segunda cosa que vamos a hacer es en el pooledDataSource vamos a poner el setUser. Y en el user, vamos a escribir la string ("root"). Y por último, en el pooledDataSource vamos a poner un setPassword. Y el Password va a ser el ("root1234").

[04:09] Inicializamos este objeto, ¿pero qué hacemos con él ahora? Bueno, ahora lo que vamos a hacer es en la connectionFactory nosotros vamos a crear

un atributo privado llamado DataSource de la clase java.sql y acá datasource también. Y nosotros vamos a asignar este pooledDataSource a este atributo.

[04:34] Entonces this.datasource = pooledDataSource, ahí está. Y por último, nosotros vamos a cambiar esta implementación de recuperar conexión y en lugar de estar utilizando todo lo que hacíamos de JDBC, nosotros vamos a estar haciendo, borrando esto y vamos a dar un return this.datasource.getConnection(); por fin vamos a organizar los importes que no estamos más utilizando y listo.

[05:04] Cambiamos, agregamos dos dependencias y cambiamos una sola clase y ya podemos ver que el proyecto no se rompió. Vamos a ejecutar la aplicación para ver si las cosas siguen funcionando. Ahí voy a levantar la aplicación. Bueno, ahí está la pantalla con todos los productos listados y todo sigue funcionando normalmente.

[05:23] La única diferencia es que en lugar de estar creando una conexión por el driver manager, nosotros estamos tomando la conexión desde el pool para realizar las operaciones en la base de datos. Y para complementar la utilización de las dependencias que agregamos, podemos ver acá en la consola, ahí voy a agrandar un poquito más.

[05:43] Podemos ver acá en la consola que tenemos las informaciones sobre la instanciación del pool de conexiones. Acá podemos ver que en el Initializing c3p0 pool tenemos un montón de configuraciones: acquireIncrement, acquireRetryDelay, CommitOnClose, hay un montón de configuraciones acá y yo les recomiendo leer la documentación de C3P0 para entender qué hace cada una.

[06:13] Les recomiendo leer la documentación. ¿Por qué? Porque acá, para no salir mucho del objetivo del curso, nosotros vamos a enfocarnos en algunas configuraciones que son clave. Entonces ya configuramos acá el pool de conexiones, voy a agrandar acá la pantalla. Ahora sí.

[06:29] Ya las configuramos acá y para la próxima clase vamos a ver más características que podemos configurar como la cantidad máxima y la cantidad mínima de conexiones que este pool puede mantener abiertas. Y también tenemos que ver qué va a pasar si necesitamos consumir más conexiones que las que están en el pool. Nos vemos en la próxima clase